**AI CUP 2024 秋季賽**

**根據區域微氣候資料預測發電量競賽**

**報告說明文件**

* 參賽隊伍需詳細說明系統的流程、演算法、工具與外部資源等，特別是創新性(如改了外部資源的哪一部分)。主辦單位會請專業學者、專家仔細審查，若發現分數有問題或方法說明不清之部分，請參賽隊伍補充說明，經發現有違規者，將取消獎項資格。
* 本競賽未禁止使用生成式AI，惟於競賽及撰寫報告過程如有使用生成式  
  AI，請如實揭露，並說明作法。
* 請使用A4紙直式打字，中文字體使用標楷體，英文、數字與符號使用Times New Roman字體。
* 版面設定：邊界上下各2.54CM，左邊3.17CM、右邊3.0CM。
* 字體大小：題目20（粗體）**、**內文12，單行間距。
* 作者聯絡資料表：詳細資料請填寫於附件的「作者聯絡資料表」。
* 須依照以下大綱及內容說明撰寫。不可自訂標題、修改內容順序，或合併段落。但可搭配圖片說明。
* 有意爭取創意獎的隊伍，可參考「[優良報告範本](https://drive.google.com/drive/folders/1_s4S-DVIs0pSkXC1lCyPdFpaWCs3uJvC?usp=share_link)」，惟撰寫格式仍應遵照下方規定。
* 附件空白報告範本可參用。
* **報告大綱與撰寫規定（字數不得低於各段落規定下限）**

壹、環境

字數規定：200~600字。

內容規定：請說明使用的作業系統、語言、套件(函式庫)、預訓練模型、額外資料集等。如使用預訓練模型及額外資料集，請逐一列出來源。

貳、演算方法與模型架構

字數規定：400~1200字。

內容規定：說明演算法設計、模型架構與模型參數，包括可能使用的特殊處理方式。

參、創新性

字數規定：300~1200字。

內容規定：說明演算法之創新性或者修改外部資源的哪一部分。

肆、資料處理

字數規定：300~1500字。

內容規定：說明對資料的處理或擴增的方式，例如對資料可能的刪減、更正或增補。

伍、訓練方式

字數規定：400~1000字

內容規定：說明模型的訓練方法與過程。

陸、分析與結論

字數規定：400~2500字。

內容規定：分析所使用的模型及其成效，簡述未來可能改進的方向。分析必須附圖，可將幾個成功的和失敗的例子附上並說明之。

柒、程式碼(未於**2024/12/09 23:59前**繳程式碼連結者，將失去獲頒獎金/獎狀資格)

內容規定：請在此提供程式碼下載連結並請另外信件附檔，包含資料處理、訓練流程、預測等相關程式碼。程式碼應附README.md檔案交代安裝配置環境，重要模塊輸出/輸入，以讓第三方用戶可以除錯、重新訓練與重現結果。繳交前請確認連結有效且有開啟瀏覽權限，如連結失效視同未交。

捌、使用的外部資源與參考文獻

內容規定：參考文獻請以APA格式為主。

★註1：請確認上述資料與AI CUP報名系統中填寫之內容相同。自2023年起，獎狀製作將依據報名系統中填寫內容為準，有特殊狀況需修正者，請主動於報告繳交期限內來信moe.ai.ncu@gmail.com。，報告繳交截止時間後將不予修改。

★註2：繳交程式碼檔案與報告，請Email至：ailabailab5051@gmail.com，並同時副本至：t\_brain@trendmicro.com與moe.ai.ncu@gmail.com。缺一不可。報告檔名與信件主旨請寫「競賽報告與程式碼/TEAM\_？？？？/根據區域微氣候資料預測發電量競賽」

**AI CUP 2024 秋季賽**

附件

**根據區域微氣候資料預測發電量競賽報告**（空白範本）

隊伍：TEAM\_6894

隊員：沈奕至(隊長)、張鳳愷。

Private leaderboard：635905.5/Rank 23

1. **環境(200~600字)**

**作業系統**：Windows 11 家用版 23H2

**程式語言與套件**：

* **Python** : 3.12.7
* **Jupter套件**：IPython(8.27.0)、ipykernel(6.29.5)、ipywidgets(8.1.2)、jupyter\_client(8.6.0)、jupyter\_core(5.7.2)、jupyter\_server(2.14.1)、jupyterlab(4.2.5)、nbclient(0.8.0)、nbconvert(7.16.4)、nbformat(5.10.4)、notebook(7.2.2)、qtconsole(5.6.0)、traitlets(5.14.3)

在這次的比賽中，我們這組皆用jupyter notebook執行相關程式，包括分析、資料前處理、model訓練等。相較於用vscode .py檔案執行，這種方式比較適合用於做分析、debug。

* **相關import套件：**numpy(1.26.4)、matplotlib(3.9.2)、seaborn(0.13.2s)、pandas(2.2.3)、scikit-learn(1.5.2)、xgboost(2.1.2)、scipy(1.14.1)

1. Numpy用於特定的運算及統一data type以利於後面的運算等。
2. 我們使用matploblib中的pyplot及seaborn用於數據分析。
3. pandas用於讀取csv檔及部分csv操作。
4. scikit-learn中，我們使用了Ridge、PolynomialFeatures、RobustScaler以及部份metrics套件。
5. xgboost則是使用裡面的XGBRegressor當作模型使用。

**額外資料集(後面會有更詳細的說明)**：

* **中央氣象局歷史資料(更新頻率 1Hr)**

關於額外資料集，考慮到Arduino設備容易因為硬體的不同而產生不同的特徵數值，所以我們使用了中央氣象局歷史的資料(2024年1月1日~ 2024年10月31日，更新頻率為1Hr)。

[466990 花蓮站](<https://codis.cwa.gov.tw/StationData>)。

* **Google Earth**

在競賽的說明pdf檔中，有提到不同偵測站的位置及樓層高度等，這些也能夠當作features做使用。

[Google Earth](<https://earth.google.com/web/search>)

1. **演算方法與模型架構(400~1200字)**

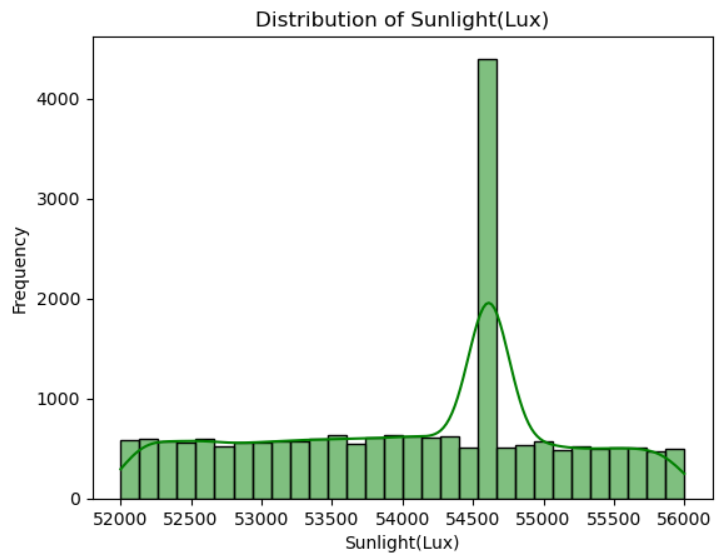
**設計流程**：

1. 資料處理: Sunlight(Lux)異常校正、Max\_Sunlight(Lux)校正
2. 預測Sunlight(Lux)
3. 預測Power(mW)

**資料處理**：

1. Sunlight(Lux)異常校正：

在進行數據分析的時候，我們發現Sunlight(Lux)在數值為54612.5時，擁有較多數量的資料(圖1)。並且在Sunlight(Lux)-Power(mW)的分布圖中，可以發現在此Sunlight(Lux)數值下的Power(mW)分布範圍幾乎涵蓋了全部範圍，推測為異常數值。

<圖1.>

參數、模型：

因為選用了數據較為正常的LocationCode進行校正，並且只是為了校正異常值，所以可以使用較強且非線性的Regressor Model。

params= {'colsample\_bytree': 0.95, 'learning\_rate': 0.3, 'max\_depth': 12, 'n\_estimators': 450, 'subsample': 0.95}，使用XGBRegressor(\*\*params)進行校正。

1. Max\_Sunlight(Lux)校正：

在競賽資料的pdf檔案中有提到，因為硬體上的限制，Sunlight(Lux)欄位最大數值只到達117758.2，所以我們需要將其恢復到應有的數值上。

為了超出數值的上限，我們使用線性的Regressor進行填補，因為線性的Regressor容易受到資料的分布而影響最終的預測結果，所以我們需要選擇數據較為正常的LocationCode進行訓練、校正，再將剩餘的資料做校正。

參數、模型：

呈1.，取那些LocationCode為1、2、5、7且Sunlight(Lux)不為最大值(117758.2)的資料，先做PolynomialFeatures(degree=3)再用RoubustScaler做scaling，最後用Ridge預測Sunlight(Lux)。

最後，再將已經校正好的資料(LocationCode 1、2、5、7)作為train\_data將其餘的LocationCode重新做預測、重建。

params = {'colsample\_bytree': 0.95, 'learning\_rate': 0.1, 'max\_depth': 12, 'n\_estimators': 450, 'subsample':0.95}，使用XGBRegressor(\*\*params)。

**預測Sunlight(Lux)** :

由於test\_data中並沒有Sunlight(Lux)的資訊，所以我們需要先用既有的 資料去預測。

params\_sunlight = {'colsample\_bytree': 0.95, 'learning\_rate': 0.3, 'max\_depth': 12, 'n\_estimators': 350, 'subsample':0.95}， XGBRegressor(\*\*params\_sunlight, objective='reg:squarederror')

**預測Power(mW)** :

將既有的資料與預測完的Sunlight(Lux)再拿去預測power。

params\_power = {'colsample\_bytree': 1.0, 'learning\_rate': 0.1, 'max\_depth': 14, 'n\_estimators': 300}

XGBRegressor(\*\*params\_power, objective='reg:absoluteerror')

1. **創新性(300~1200字)**

說明演算法之創新性或者修改外部資源的哪一部分

在競賽說明文件中有提供每個資料蒐集站的粗略位子，我們能夠經由該表所提供的資訊，透過Google Earth的座標位置查詢，取得更多資訊。在本次比賽中，我們新增了位置、面朝向、高度等資訊。

關於特徵Sunlight(Lux)，競賽說明文件中也有提到其對於Power(mW)具有極高的解釋度，然而裡面仍然有許多躁點需要做修正。與其直接取平均值做填補，我們選擇將資料分布較為正常的幾個測量站當作training data，再由這些資料去重建其餘測量站的Sunlight(Lux)。

另外在選擇重建的training data上除了選擇數據較為穩定的測量站外，還可以細分不同月份不同測量站的數據穩定性，如此就可以使得Sunlight(Lux)的數據更為穩定。

在重建Max\_Sunligh的模型選擇上，我們捨棄了使用XGBRegressor這個較為常見且強大的Regressor Model，因為他只能在特定範圍內達到精確度佳的表現，然而我們的目的是重新建立出Max\_Sunlight點的實際Sunlight數值，所以我們最終使用了Ridge此Regressor演算法。

除了模型選擇外，為了避免Sunlight模型預測出來的數值有負數的情況，在使用XGBRegressor前，都有把Sunlight進行np.log1p的操作，最後再用np.expm1做復原；在使用Ridge的時候我們並沒有這樣做，因為XGBRegressor較為強大，可以有效的區分絕大多數的數值，但是Ridge預測出來的數值通常有較多的誤差，這會導致最終的結果誤差較大。

1. **資料處理(300~1500字)**

由比賽給的Train Dataset及Upload等csv檔可以得知，在Upload中並沒有Train Dataset所給的特徵值，也就是我們需要自己去做填補，亦或是捨棄這些特徵。除此之外，我們在新增特徵的同時，也需要考量Upload的資料是否也能夠做填補。

我們的選擇是將Train Dataset中的所有特徵都捨棄，並且換成中央氣象局的歷史數據。儘管中央氣象局的資料是以每小時的頻率進行更新，但考量到不同測量站的硬體設備表現可能各不相同(即便是使用相同牌子的硬體設備，仍然可能有些許差異)，這可能導致最終Model的預測受到干擾。

此外，我們也透過Google Earth針對不同偵測站新增了位置、面朝向、高度等資訊。位置區分了東華校區與花蓮氣象站附近的偵測站；面朝向代表太陽能發電板的面朝向；高度是參考該區域的海拔再加上樓層高\*(樓層數-1)，為了方便樓層高我們設為4.2m。

我們針對Sunlight的異常值(54612.5) 做處理，首先將偵測站為1、2、5、7的資料取出，剔除異常值及最大值(117758.2)後，並且使用這些資料訓練XGBRegressor，之後將1、2、5、7偵測站的異常值還原。此外，為了避免Sunlight最大值影響最終的模型預測，我們再次取出偵測站為1、2、5、7的資料，剔除最大值，並依序做PolynomialFeature、Standardize，然後使用Ridge進行預測。最後，利用已經完全修復好的1、2、5、7偵測站資料去修復其餘的偵測站。

**伍、訓練方式(400~1000字)**

所有的模型訓練都是將data分成8:2的比例，8用來當作train data，2用來當作validation data。除此之外，考慮到數值的分布不均勻(如圖2所示)，我們使用numpy將數據進行分段，並且使用stratify的方式，讓split後的數據分布能夠更加均勻，避免影響模型訓練。

一張含有 文字, 圖表, 螢幕擷取畫面, 繪圖 的圖片

自動產生的描述<圖2.>

在所有有關Sunlight(Lux)預測、修正的model中，我們皆使用Mean Squard Error當作loss function，因為在訓練之前，我們選擇的對象的outliers相對較少，這能夠有效的使模型快速收斂。在最終的Power(mW)預測模型中使用Mean Absolute Error當作loss function，單純只是想要與比賽的評分方式相同。

在挑選最佳參數上，在比賽的前幾天我們使用RandomizedSearchCV進行參數挑選，這大概會花費我們2個小時左右進行參數挑選。然而，最好的成績是在最後一天產生的，當天我們是用類似先前的參數進行訓練，並且透過調整XGBRegressor的max\_depth、n\_estimators選擇最佳loss數值的參數。由於硬體上的限制，我們沒辦法很快速的訓練模型，但是我們仍然推薦使用RandomizedSearchCV或GridSearchCV進行參數挑選。

說明模型的訓練方法與過程。

1. **分析與結論(400~2500字)**

分析所使用的模型及其成效，簡述未來可能改進的方向。分析必須附圖，可將幾個成功的和失敗的例子附上並說明之。

**柒、程式碼**

Github連結：

**捌、使用的外部資源與參考文獻**

**作者聯絡資料表**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 隊伍名稱 |  | Private Leaderboard 成績 |  | Private Leaderboard 名次 |  |
| 身分  （隊長/隊員） | 姓名 (中英皆需填寫)  (英文寫法為名,姓，例：Xiao－Ming, Wu，名須加連字號，姓前須加逗號) | 學校＋系所中文全稱  (請填寫完整全名，勿縮寫) | 學校＋系所英文中文全稱  (請填寫完整全名，勿縮寫) | 電話 | E-mail |
| 隊長 | (範例)  吳小明  Xiao－Ming, Wu | 國立中央大學資訊工程學系 | National Central University Department of Computer Science & Information Engineering | 0900-123-456 | abc＠gmail.com |
| 隊員1 |  |  |  |  |  |
| 隊員2 |  |  |  |  |  |
| 隊員3 |  |  |  |  |  |
| 隊員4 |  |  |  |  |  |

★註1：請確認上述資料與AI CUP報名系統中填寫之內容相同。自2023年起，獎狀製作將依據報名系統中填寫內容為準，有特殊狀況需修正者，請主動於報告繳交期限內來信moe.ai.ncu@gmail.com。報告繳交截止時間後將不予修改。

★註2：繳交程式碼檔案與報告，請Email至：ailabailab5051@gmail.com，並同時副本至：t\_brain@trendmicro.com與moe.ai.ncu@gmail.com。缺一不可。